REMARKS

In the final rejection dated June 6, 2005, the Examiner maintained his rejection of most claims as either anticipated by the patent to Troung, or obvious in view of the combination of Truong and D'Arlach.[1] However, applicant has a fundamental disagreement with the Examiner regarding the scope and content of the cited prior art with regard at least two basic issues, and this will be applicant's final, abbreviated attempt before appeal to convince the Examiner that he is wrong about at least these two issues as it relates to claims 1 and 6. Other claims have similar language and would also be patentable if applicant's reasoning convinces the Examiner, but for simplicity, applicant has limited this response to a discussion of claims 1 and 6.

I. EDITOR

The first fundamental disagreement relates to scope and content of D'Arlach's disclosure of an editor. Most simply, applicant asserts that D'Arlach does not teach or suggest "*an editor directly operating on pages displayed by the browser via the editing features, thereby allowing the user to work on a **functional** application during development,*" as recited in claim 1. The Examiner disagrees, noting that

> D'Arlach does disclose in analogous art, creating or editing a working copy of a user's site with the option of publishing the updated modified page or creating a new user web site. Therefore it would have been obvious . . . to combine Troung and D'Arlach because, it would enable a working copy or user web page to be modified dynamically.

(*See* Office Action dated 6/6/05 at p.4, *citing* D'Arlach at col. 5:15-25). It appears that the Examiner has improperly latched onto the term "working copy" as used in D'Arlach to support and justify his conclusion. However, the "working copy" described in D'Arlach is clearly not the "*functional application*" referred to in applicant's claims, and for this reason, the Examiner's logic must fail. A "working copy" means a copy for the user to work on, not a copy that works as a running and functional application. This interpretation is supported by the disclosure of D'Arlach, which says that after changes to the web site are completed, the user may publish those changes. (*See* D'Arlach at col. 5:15-25):

[1] Claims 22-24, 25-27, 30-33, 43 and 67 stand rejected as anticipated by Troung; claims 1-8, 28-29, 41-42, 51-66, 68-96 and 114-127 stand rejected as obvious over the combination of Troung and D'Arlach; and claim 25 stands rejected as obvious over the combination of Troung, D'Arlach and Popp.

> *To allow creating a new Web site a CGI program first makes a copy of an exiting template in the server computer. The user then customizes or edits the* ***working*** *copy of the template, which is the user's site, through a series of forms displayed by the browser in the client computer. After making desired changes to the site, the user may publish it.*

Thus, D'Arlach's editor is only useful for editing a static web site, not for editing a functional, running web application, and it does not provide any teaching or suggestion of any way to dynamically modify a functional, running web application.

Server based dynamic web applications dynamically generate web pages while a user visits the application. Web sites, on the other hand, do not have generation taking place on the server while a user visits the site. *See*, for example, D'Arlach at col. 5:30-33:

> *When the template is published as a Web site, the database is used to generate a set of Web pages that make up the new site.*

This makes clear that the generation in D'Arlach takes place when the web site is published, and not later when users visit the site. Thus, D'Arlach is not suited for developing dynamic server based web applications. In contrast, applicant's invention is advantageous in that a user can edit server side web applications while the application appears and functions normally.

The Examiner also mentions the language from D'Arlach that editing is accomplished "*through a series of forms displayed by the browser.*" This statement refers to the process of editing (how the editor works), but it does not say anything about the item being edited (*the working copy of the template*) and is thus irrelevant.

II. COMPONENTS

The second fundamental disagreement relates to the scope and content of the combined disclosures of D'Arlach and Truong with regard to components.

On pp. 16-17 of the current action, the Examiner states that the "components" recited in applicants claim 6 are equivalent to the objects described in D'Arlach, and that nothing in applicant's claims distinguishes the claimed components from the objects described in D'Arlach. However, applicant's claim 6 requires "*at least one component that reacts interactively on user input by executing instructions contained in said component on the server*". Applicant had previously amended claim 6 to add the limitation that the instructions to be executed were "*contained in said component*" but the Examiner never commented on this amendment and appears to still base his argument on an old version of the claim.

Further, the Examiner does not explicitly argue that the elements described in D'Arlach contain instructions or are executable. However, on page 16 of the current action, the Examiner disagrees with applicant's assertion that D'Arlachs templates do not have **software** components. The Examiner further states that D'Arlach's templates are customizable and contain objects and elements which can be customized. Being customizable, however, does not teach or suggest that an object or element would contain instructions that are executable.

On page 16 of the action, the Examiner further argues that storing elements in databases does not matter. However, since databases are commonly used to store data and not executable instructions, the choice of D'Arlach to use a database for his templates indicates that he sees templates and elements as data, rather than as programs. In fact, the database structure described by D'Arlach (*see* col. 5:1-15) does make any reference to instructions for execution on the server nor executable components. Further, D'Arlach explicitly describes only two kinds of elements: a text element and a button element, and neither contains instructions for execution on the server.

On page 17 of the action, the Examiner gives further support to applicant's position by citing examples of elements, namely graphical icons, text boxes or buttons, and he goes on to state that java beans, in contrast, are different. Applicant submits that the reason that java beans are different, like applicant's components, is because they contain instructions and therefore are executable, while the elements described in D'Arlach do not. Thus, applicant submits that "*executing instructions contained in said component*" is a distinctive feature.

III. INFORMALITIES

The office action summary indicates that claim 73 was withdrawn, which is an error. Also claim 33 was indicated both as withdrawn and rejected, which is also an error – claim 33 has not been withdrawn. Further, new claim 128 was neither rejected nor allowed.

IV. CONCLUSION

For all the foregoing reasons, applicant submits that the claims are in condition for allowance, and the Examiner's reconsideration to that end is respectfully solicited. The Examiner is encouraged to telephone the undersigned should additional issues remain.

Respectfully submitted,

Dated: 8/10/05

By:
Richard A. Nebb
Reg. No. 33,540

JEFFER, MANGELS, BUTLER & MARMARO LLP
1900 Avenue of the Stars, 7th Floor
Los Angeles, CA 90067
(415) 984 9608
mebb@jmbm.com